FortiAuthenticator - Captive Portal Guide

VERSION 1.0

FORTINET DOCUMENT LIBRARY

http://docs.fortinet.com

FORTINET VIDEO GUIDE

http://video.fortinet.com

FORTINET BLOG

https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT

https://support.fortinet.com

http://cookbook.fortinet.com/how-to-work-with-fortinet-support/

FORTIGATE COOKBOOK

http://cookbook.fortinet.com

FORTINET TRAINING SERVICES

http://www.fortinet.com/training

FORTIGUARD CENTER

http://www.fortiguard.com

END USER LICENSE AGREEMENT

http://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK

Email: techdocs@fortinet.com

09/09/2015

FortiAuthenticator 4.0 - Captive Portal Guide

23-330-264235-20150901

TABLE OF CONTENTS

Change Log

Date	Change Description
2015-09-09	Initial release.

Introduction

This document provides a general guide to Social Wifi, as an authentication method for the Guest Management feature, through Captive Portal. This document demonstrates how to configure your FortiGate and FortiAuthenticator for each social portal, and shows how a guest user can access your Wifi network without the need to register.

Instead of local authentication or remote LDAP authentication, FortiAuthenticator utilizes existing third-party identity systems to authenticate and identify users.

This solution supports:

- Facebook
- Google+
- Twitter
- LinkedIn
- Form-based authentication

Each authentication method requires the administrator to sign up to the social media website as a developer. Once signed up and logged in, you must configure the social network to allow communication with FortiAuthenticator. The steps include:

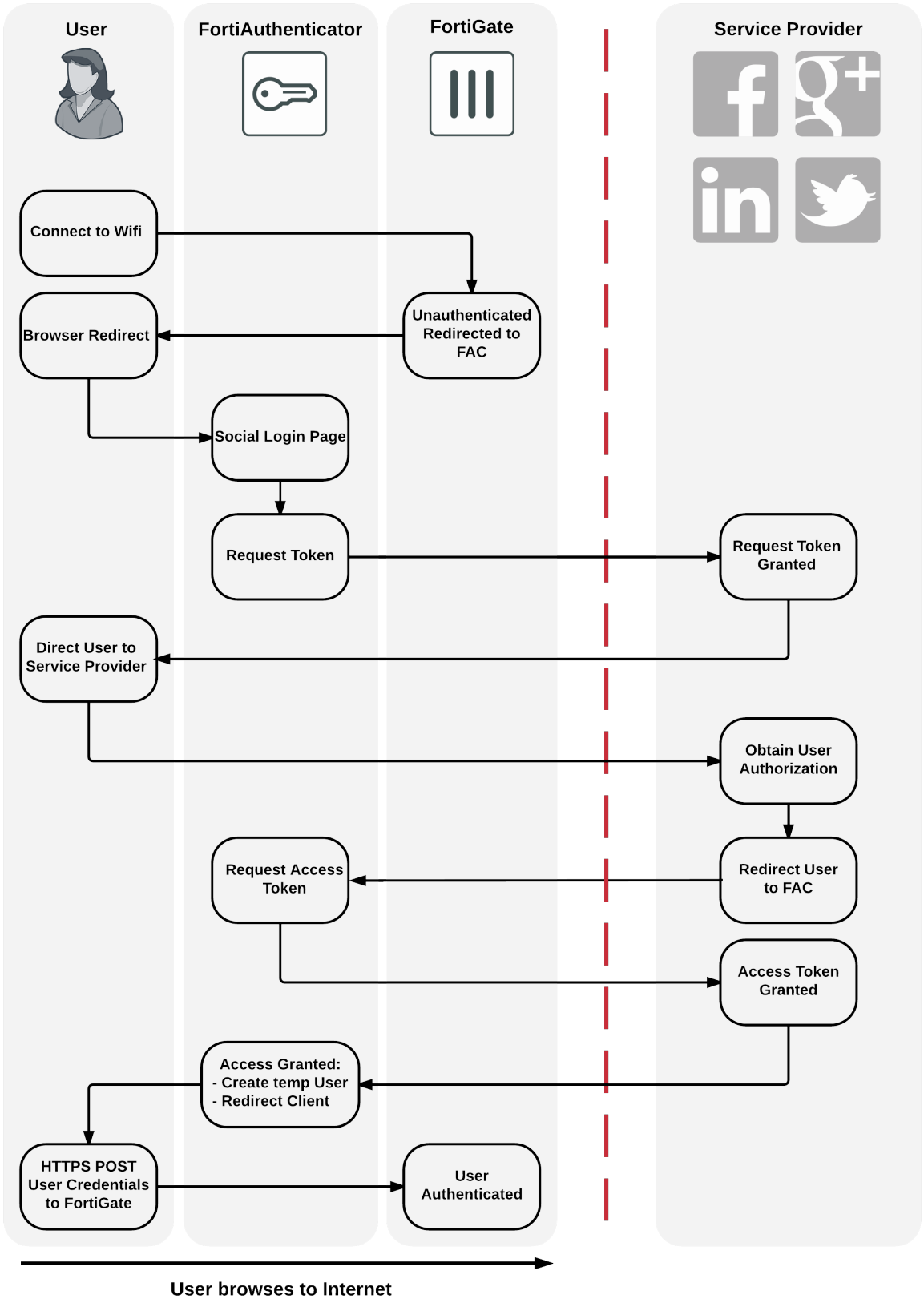
- FortiAuthenticator configuration
- FortiGate configuration

In addition, this document also provides the following information:

- Replacement Message configuration
- De-Authenticating users

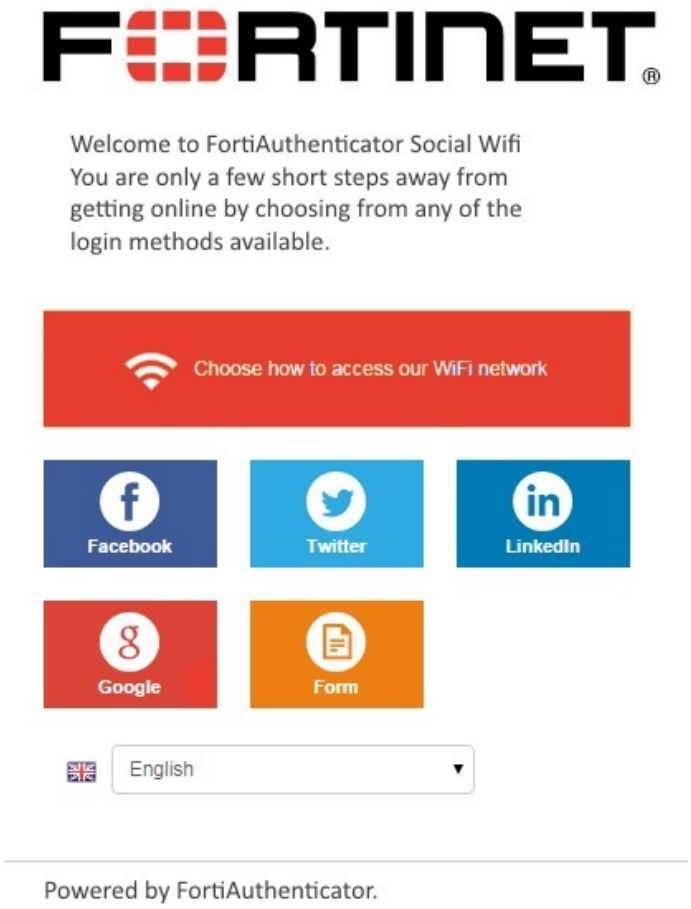
Social authentication workflow

The following diagram illustrates the process through which the user is granted social authentication:


User
FortiAuthenticator
FortiGate
Service Provider
Connect to Wifi
Unauthenticated Redirected to FAC
Browser Redirect
Social Login Page
Request Token
Request Token Granted
Direct User to Service Provider
Obtain User Authorization
Request Access Token
Redirect User to FAC
Access Token Granted
Access Granted:
- Create temp User
- Redirect Client
HTTPS POST User Credentials to FortiGate
User Authenticated
User browses to Internet

Social authentication splash page

The login (or splash) page for social authentication is configurable, allowing administrators to choose which social portals are available for their users to log in to the WiFi with. Below is the template showing all social login portals available:



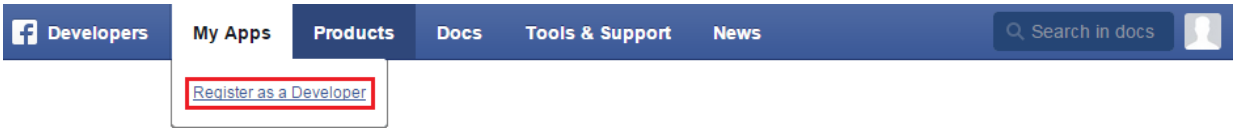
For information on customizing the splash page, see Replacement Message configuration on page 35.

Facebook

The following section demonstrates how to set up a Facebook developer account to allow social authentication through Facebook.

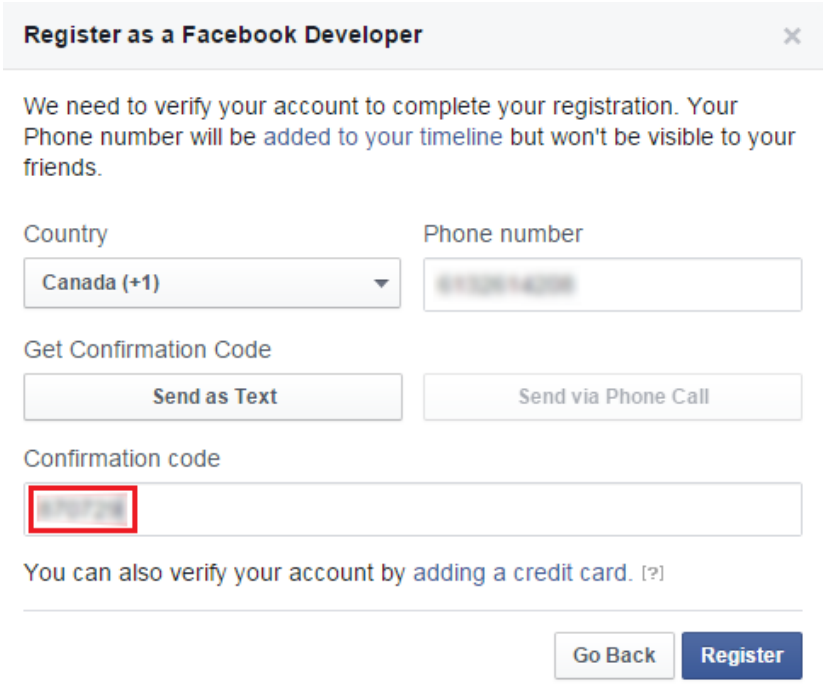
Configuring Facebook

1. Open a browser, log in to your desired Facebook account, and navigate to the following URL: https://developers.facebook.com/products/login/
   Select *My Apps* and select *Register as a Developer*.



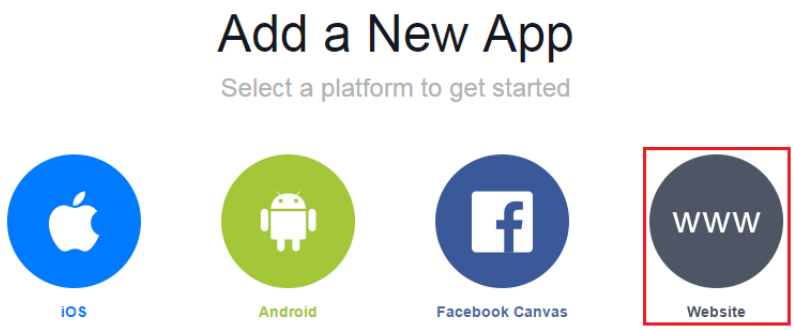
2. Enter the Facebook user's password to continue.
   You will be asked to accept the Facebook Platform and Facebook Privacy policies.
3. Select *Yes* to accept, and continue.

   Your account needs to be verified via SMS or phone call for you to continue.
4. Enter your phone number, and select to send the confirmation code as a text. Once received, enter the confirmation code, and select *Register* to continue.

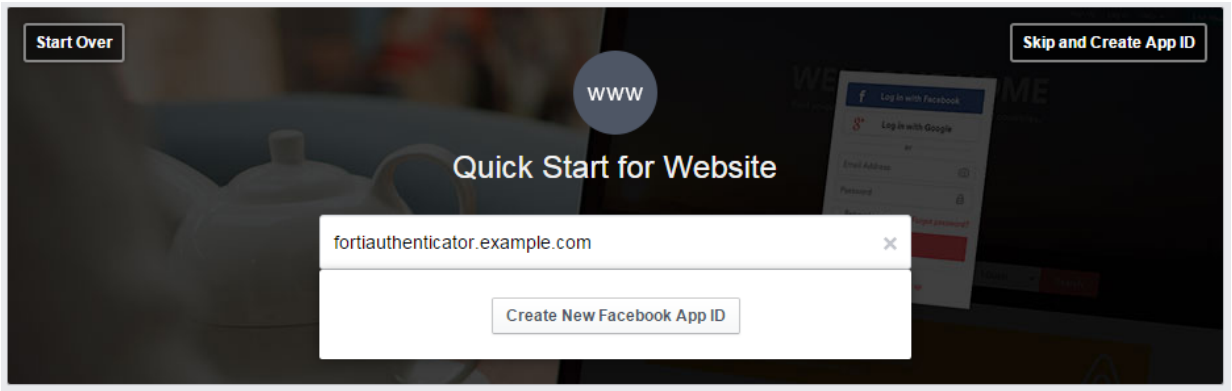


You will now be registered as a Facebook Developer.

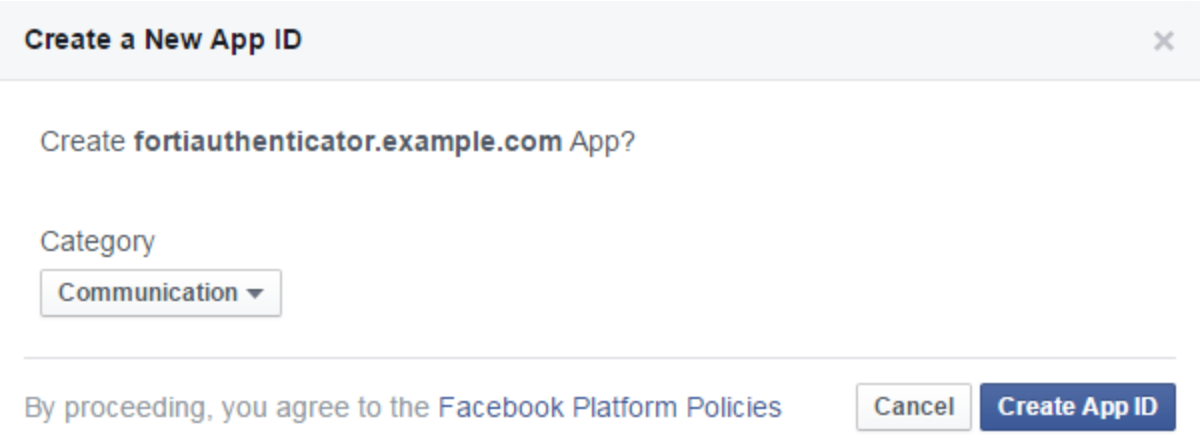
5. Select the *Website* platform to add a new app.



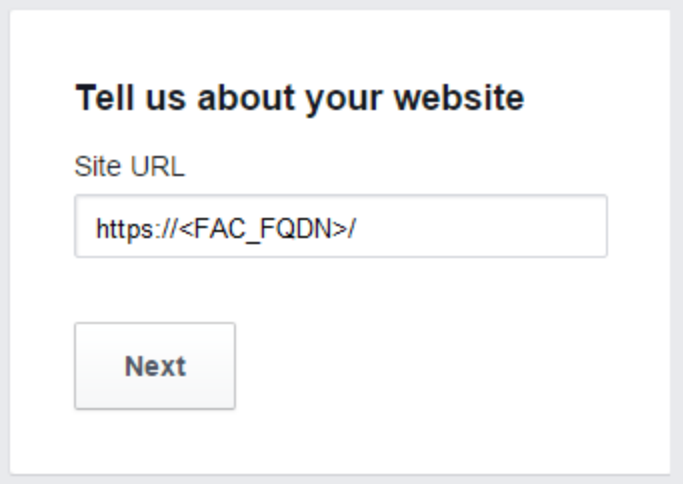
6. Enter a name for the website, and select *Create New Facebook App ID*.



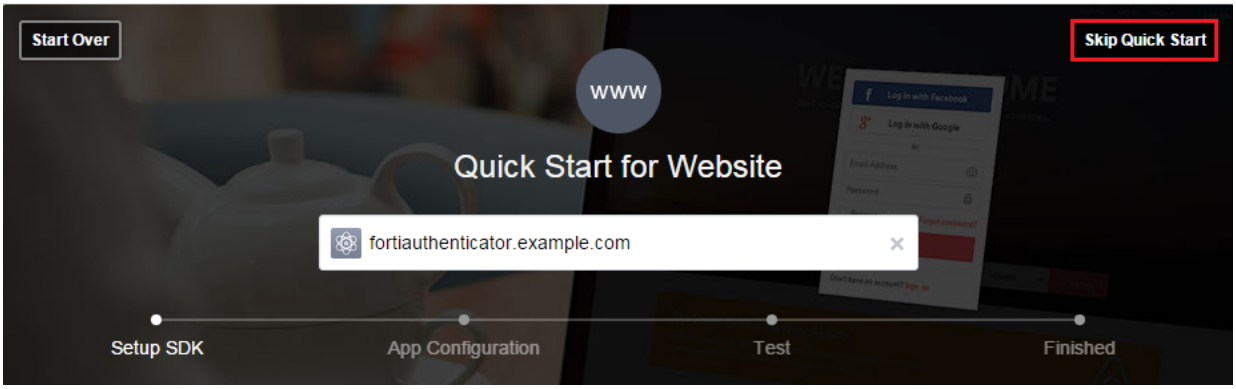
7. Select *Communication* from the dropdown *Category* menu, and select *Create App ID*.



8. Scroll down to the bottom of the page and enter the site's URL, then select *Next*.



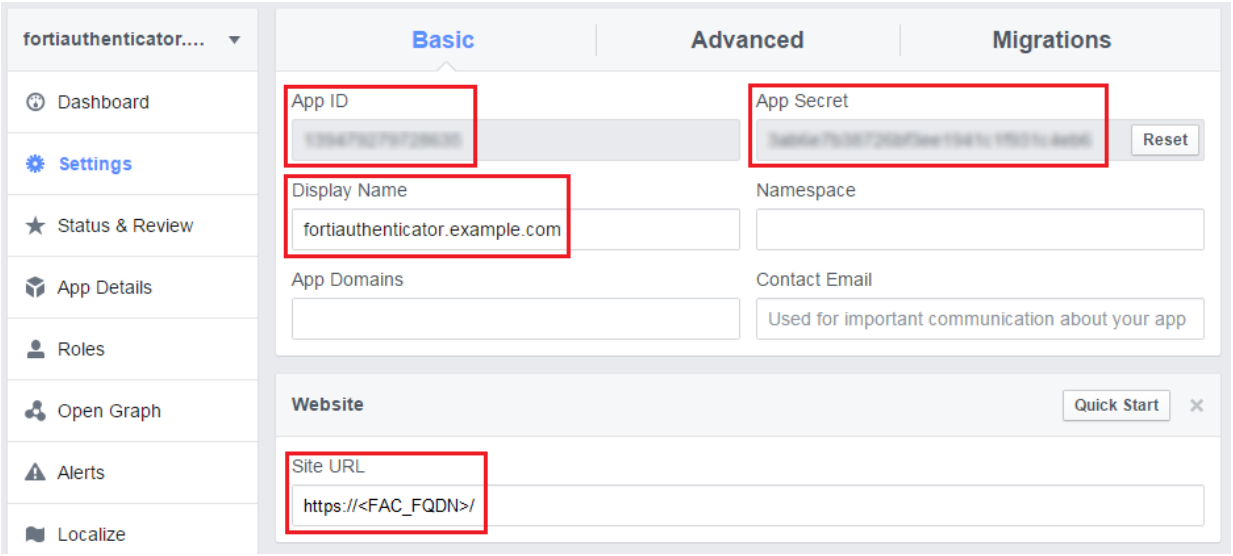
9. Scroll back up to the top of the page and select *Skip Quick Start*.



10. To confirm the configuration, go to *Settings*. From here you can see your *App ID*, *App Secret*, *Display Name*, and *Site URL*.

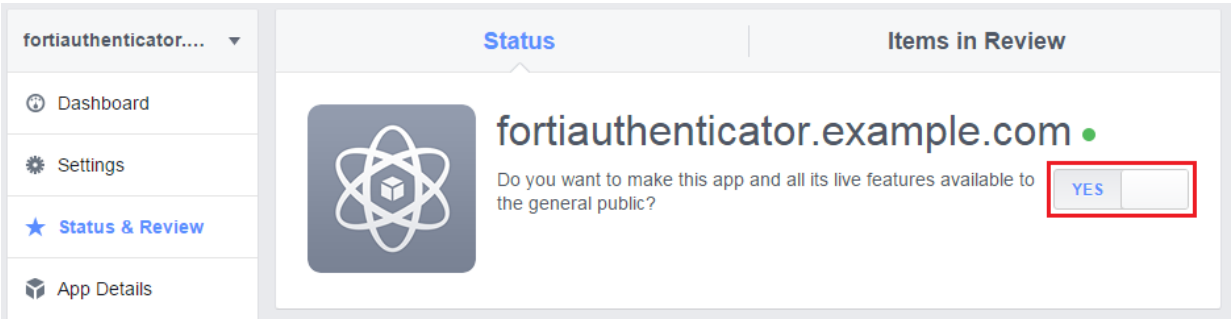
You should take note of the *App ID* and *App Secret* as it is required when configuring the Captive Portal on the FortiAuthenticator.

The *App ID* and *App Secret* can be accessed at any time on the developer account, but it may be a good idea to copy it to a secure location.



Your app needs to be 'live' before WiFi users can successfully authenticate themselves through Facebook.

11. Go to *Status & Review*, and enable the application. This will make all its features live and available to users.



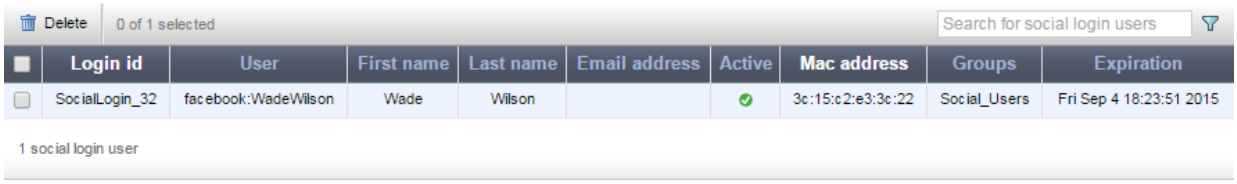
Now that the Facebook developer app is live, you can proceed to configure other social network authentication methods, or skip to FortiAuthenticator configuration on page 24.

User authentication

When the user connects to your WiFi network and navigates to a specific URL, the user is redirected to the splash page.

The user selects the Facebook portal and gets redirected to Facebook. When the user enters their credentials they are authenticated and redirected to the URL they initially requested.

To verify that the user has been authenticated on the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users*.

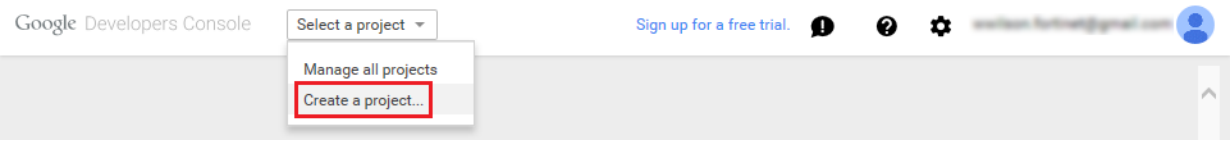

Delete
0 of 1 selected
Search for social login users
Login id
User
First name
Last name
Email address
Active
Mac address
Groups
Expiration
SocialLogin_32
facebook:WadeWilson
Wade
Wilson
3c:15:c2:e3:3c:22
Social_Users
Fri Sep 4 18:23:51 2015
1 social login user

Google+

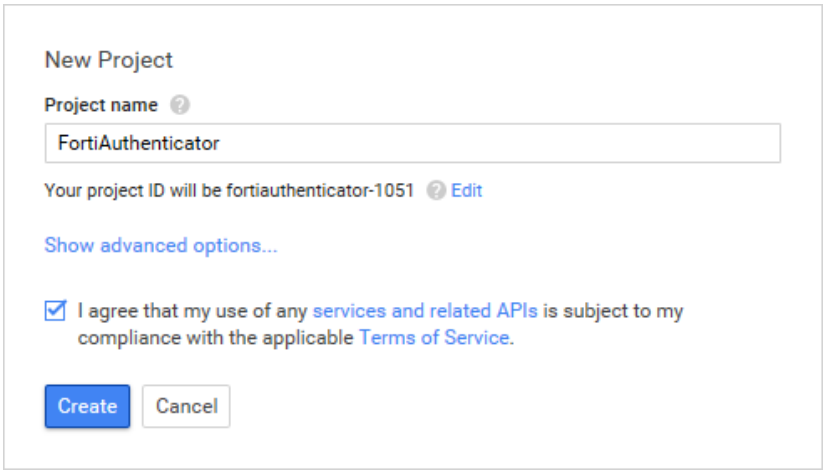
The following section demonstrates how to set up a Google+ developer account to allow social authentication through Google.

Configuring Google+

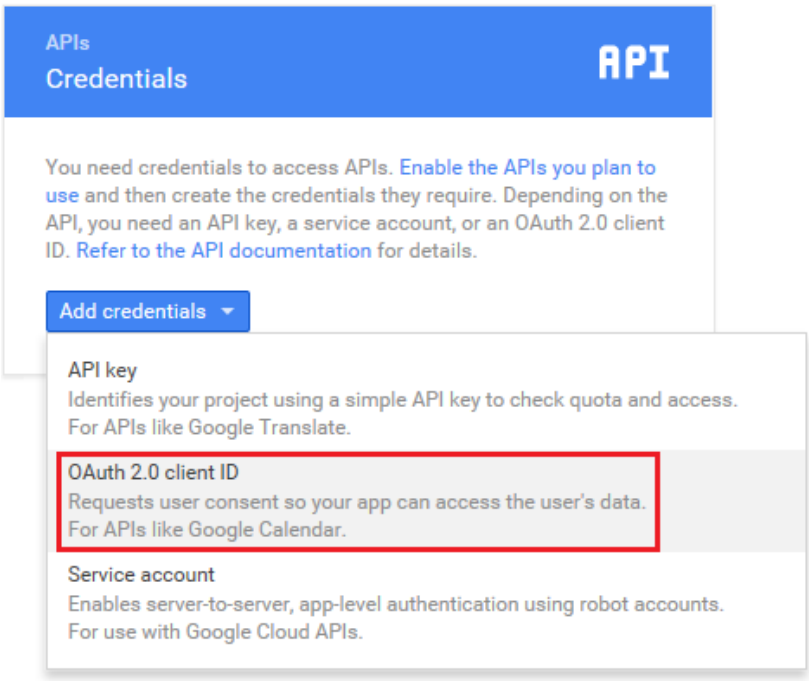
1. Open a browser, log in to your desired Google+ account, and navigate to the following URL: https://console.developers.google.com
   Open the *Select a project* dropdown and select *Create a project*.



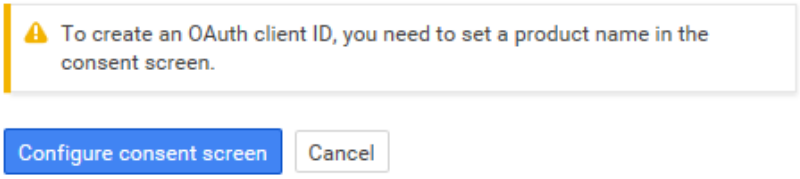
2. Enter a name for the project, accept the Terms of Service, and select *Create*.



3. Go to *APIs & auth* > *Credentials*, and select *OAuth 2.0 client ID* from the *Add credentials* dropdown.



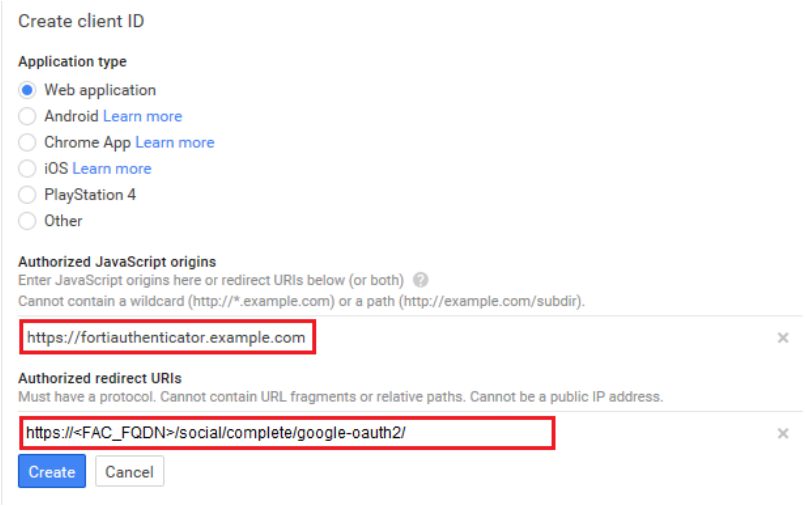
4. When prompted, select *Configure consent screen*.



5. Enter an *Email address* and *Product name* and select *Save*.
   You must now create the client ID.

6. Set *Application type* to *Web application*.
   Under *Authorized JavaScript origins*, enter the FortiAuthenticator FQDN.
   Under *Authorized redirect URIs*, enter the following:

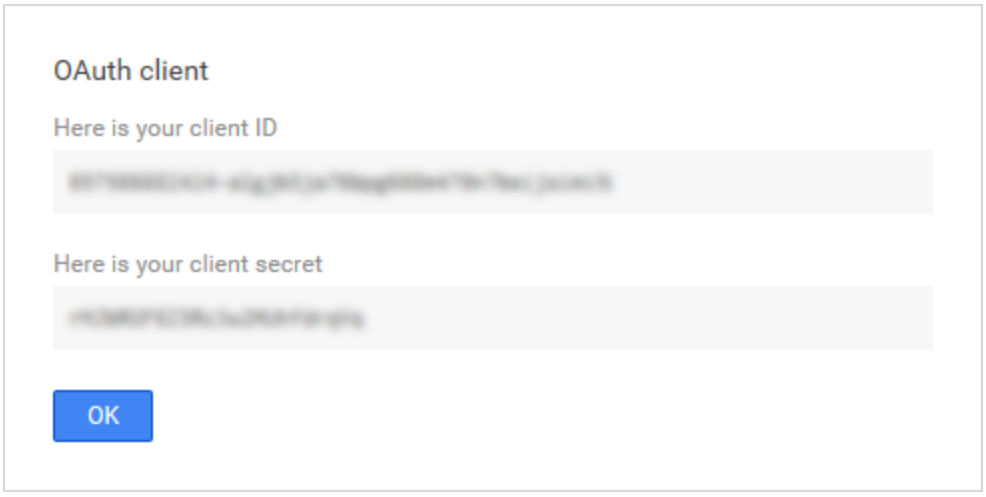
   ```
   https://<FAC_FQDN>/social/complete/google-oauth2/
   ```



7. Upon creating the client ID, a window will appear with your *client ID* and your *client secret*.

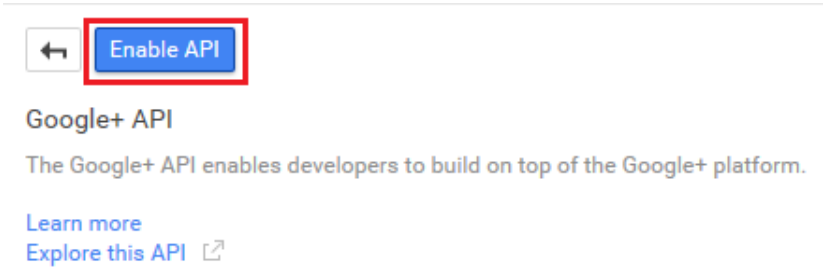
You should take note of the *client ID* and *client secret* as it is required when configuring the Captive Portal on the FortiAuthenticator.

The *client ID* and *client secret* can be accessed at any time on the developer account, but it may be a good idea to copy it to a secure location.



8. Go to *APIs & auth* > *APIs* > *Social APIs*, and select *Google+ API*.

9. Enable the API.



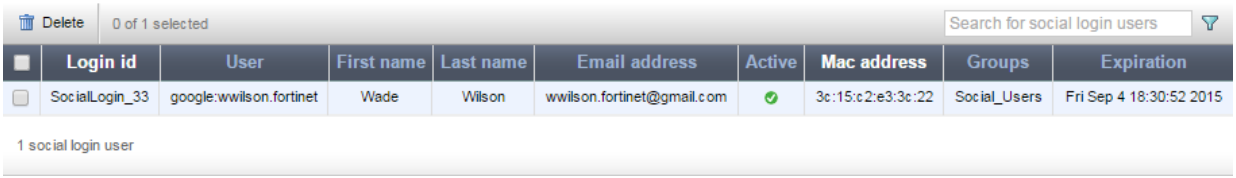
Now that the Google+ API is live, you can proceed to configure other social network authentication methods, or skip to FortiAuthenticator configuration on page 24.

User authentication

When the user connects to your WiFi network and navigates to a specific URL, the user is redirected to the splash page.

The user selects the Google portal and gets redirected to Google. When the user enters their credentials they are authenticated and redirected to the URL they initially requested.

To verify that the user has been authenticated on the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users*.

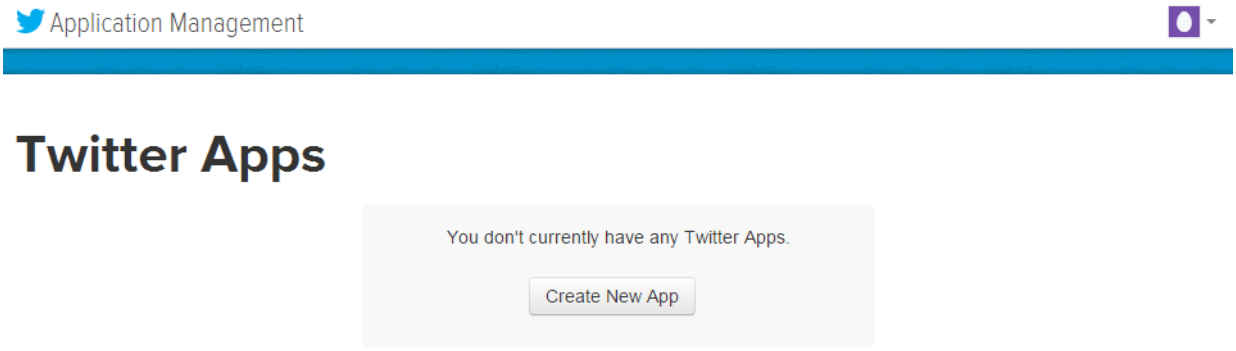


Twitter

The following section demonstrates how to set up a Twitter developer account to allow social authentication through Twitter.

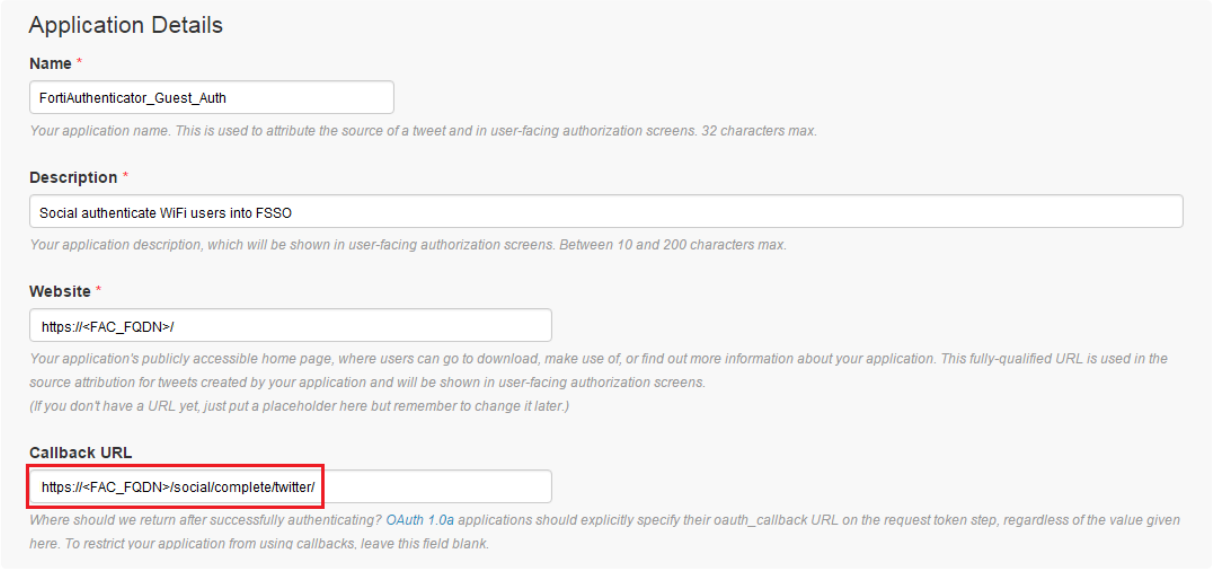
Configuring Twitter

1. Open a browser, log in to your desired Twitter account, and navigate to the following URL:
   https://apps.twitter.com/
   Select *Create New App*.

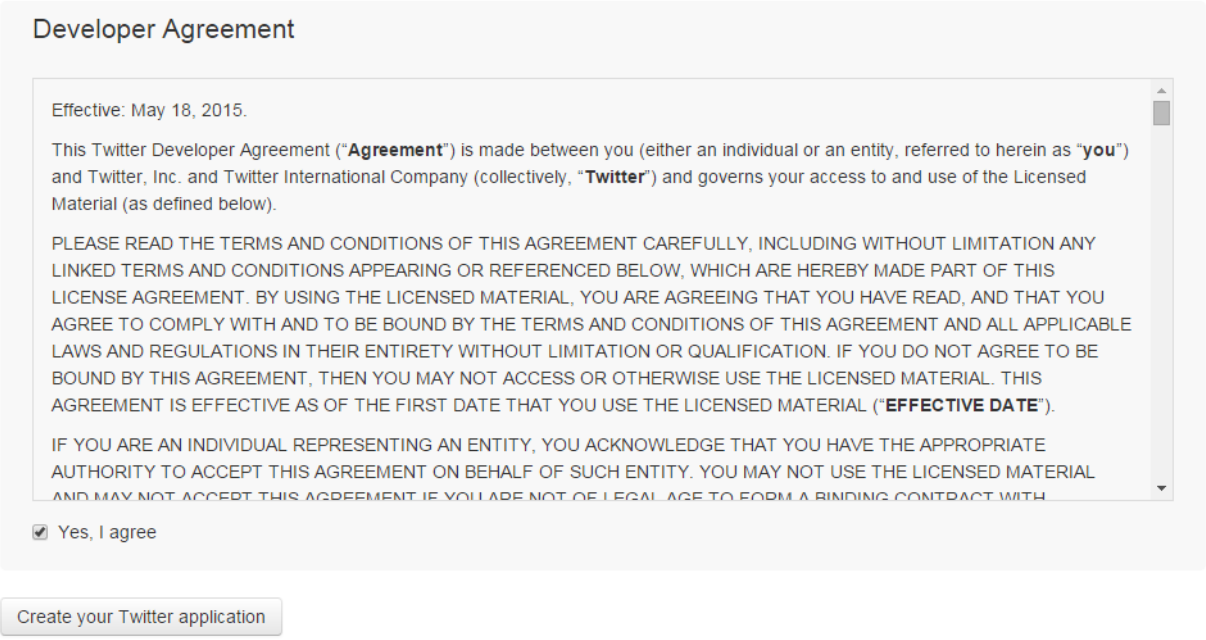


2. Enter a *Name*, *Description*, and *Website* for the application.
   In *Callback URL*, enter the following:

```
https://<FAC_FQDN>/social/complete/twitter/
```



3. Accept the *Developer Agreement* and select *Create your Twitter application*.

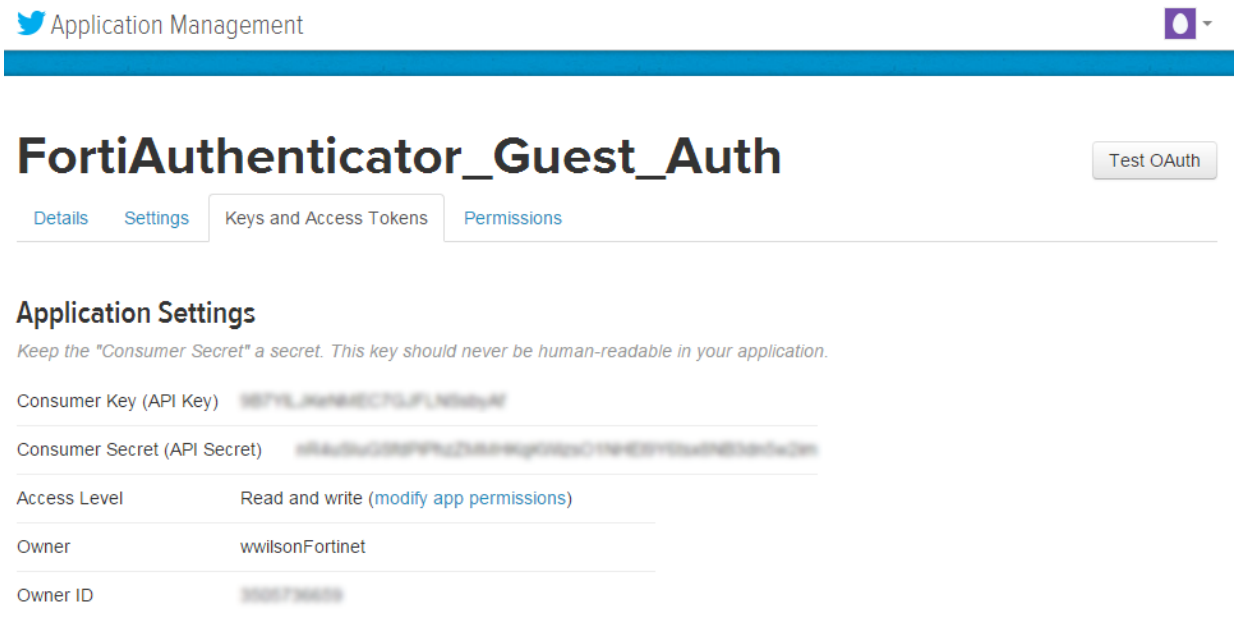


Your application has been created.

4. Go to *Keys and Access Tokens* to view your *Consumer Key* and *Consumer Secret*.

You should take note of the *Consumer Key* and *Consumer Secret* as it is required when configuring the Captive Portal on the FortiAuthenticator.

The *Consumer Key* and *Consumer Secret* can be accessed at any time on the developer account, but it may be a good idea to copy it to a secure location.



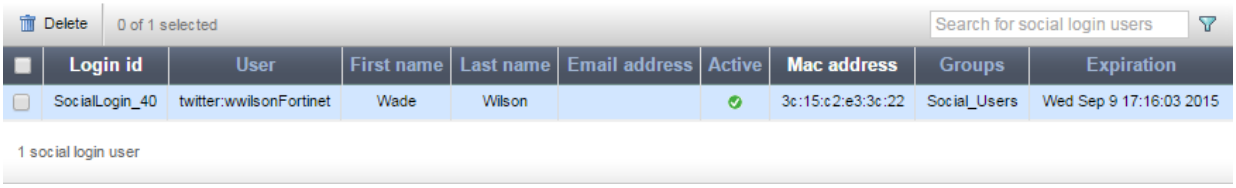
Now that the Twitter API is live, you can proceed to configure other social network authentication methods, or skip to FortiAuthenticator configuration on page 24.

User authentication

When the user connects to your WiFi network and navigates to a specific URL, the user is redirected to the splash page.

The user selects the Twitter portal and gets redirected to Twitter. When the user enters their credentials they are authenticated and redirected to the URL they initially requested.

To verify that the user has been authenticated on the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users*.

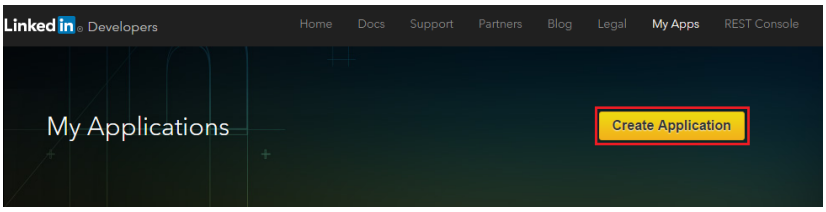


LinkedIn

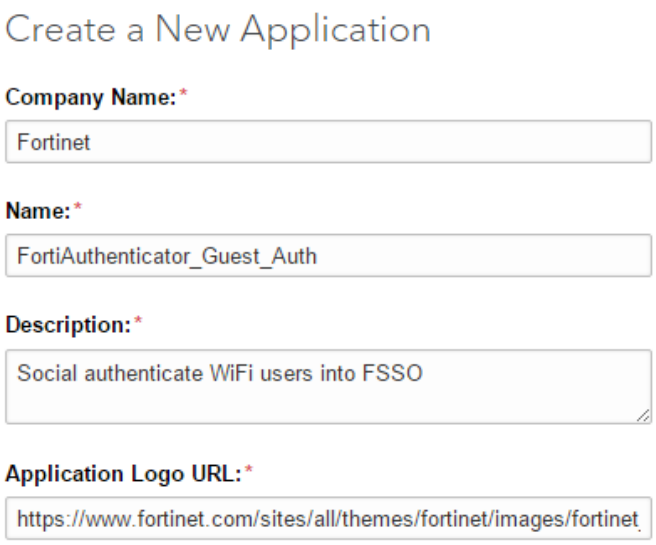
The following section demonstrates how to set up a LinkedIn developer account to allow social authentication through LinkedIn.

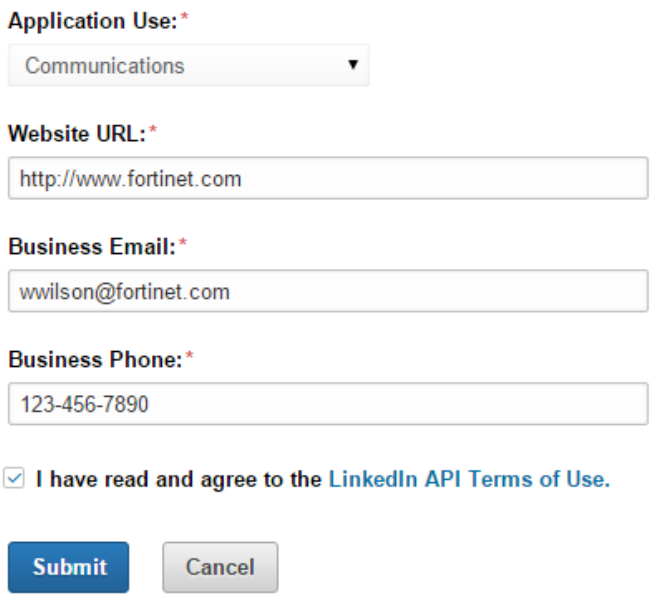
Configuring LinkedIn

1. Open a browser, log in to your desired LinkedIn account, and navigate to the following URL: https://developer.linkedin.com/documents/authentication
   Select *Create Application*.



2. Enter the new application's information in the required fields. Unlike the other social applications, LinkedIn requires an *Application Logo URL*, where you provide a link to your logo of choice; the specified logo will appear on the LinkedIn user login screen. Select that you have read and agree to the LinkedIn API Terms of Use and select *Submit*.





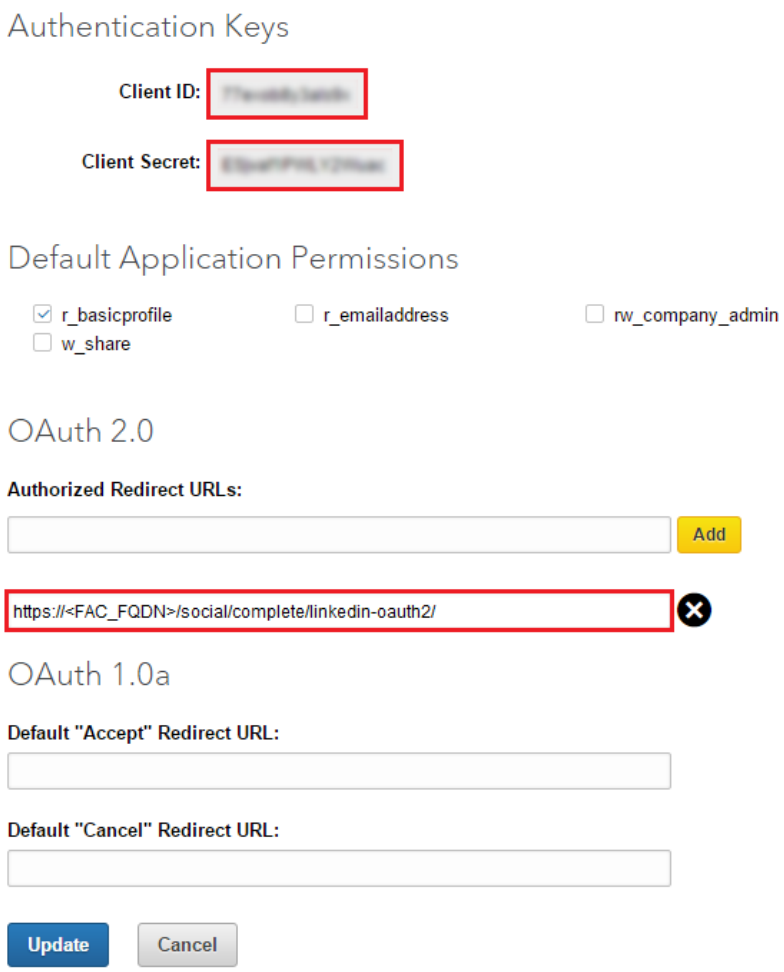
3. The next screen shows your *Client ID* and *Client Secret*.

You should take note of the *Client ID* and *Client Secret* as it is required when configuring the Captive Portal on the FortiAuthenticator.

The *Client ID* and *Client Secret* can be accessed at any time on the developer account, but it may be a good idea to copy it to a secure location.

Under *Authorized Redirect URLs*, enter the following:

```
https://<FAC_FQDN>/social/complete/linkedin-oauth2/
```



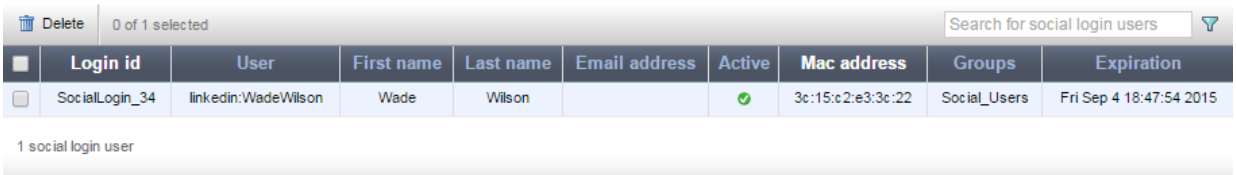
Now that the LinkedIn developer app is live, you can proceed to configure other social network authentication methods, or skip to FortiAuthenticator configuration on page 24.

User authentication

When the user connects to your WiFi network and navigates to a specific URL, the user is redirected to the splash page.

The user selects the LinkedIn portal and gets redirected to LinkedIn. When the user enters their credentials they are authenticated and redirected to the URL they initially requested.

To verify that the user has been authenticated on the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users*.



Form-based authentication

The following section demonstrates how to configure the Captive Portal for Form-based authentication on the FortiAuthenticator to allow user SMS and e-mail self-registration.

Configuring Form-based authentication

1. Go to *Authentication* > *Captive Portal* > *General*.
2. Under *Social Portal*, select *Enable SMS self-registration* and/or *Enable e-mail self-registration*. Enabling both methods will present the user with both options at the time of registration.

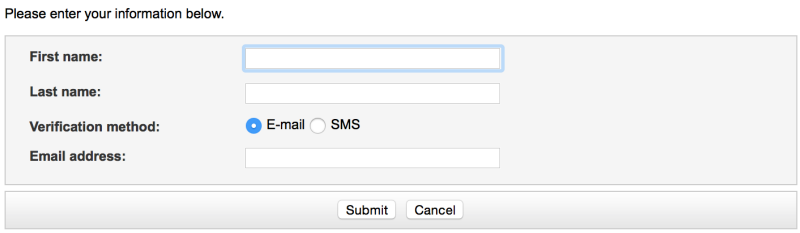
User authentication

When the user connects to your WiFi network and navigates to a specific URL, the user is redirected to the splash page. The user selects the Form portal and is prompted to enter credentials.

The user must enter their first and last name, choose a method, and enter a valid e-mail address or mobile number (including the international country calling code, e.g. +1 for North America). The field at the bottom will change depending upon which verification methods is chosen.

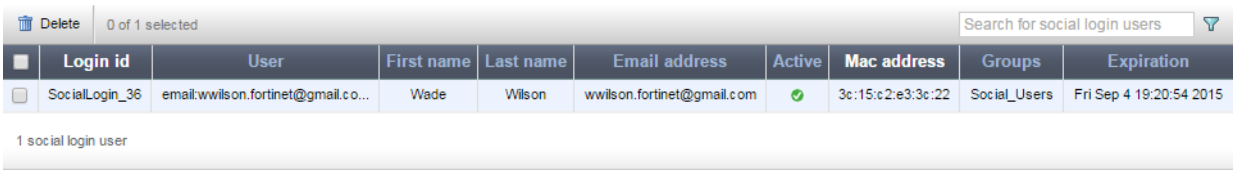
Upon selecting *Submit*, the user should receive either an e-mail or SMS text message with a six-character confirmation code, which they must enter in the *Verification code* field in order to authenticate.

The following image shows the Form-based authentication login portal, with both SMS and e-mail verification methods available:



When the user enters their credentials they are authenticated and redirected to the URL they initially requested.

To verify that the user has been authenticated on the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users*.



FortiAuthenticator configuration

Once each of the social developer accounts have been properly configured, you need to enable and configure the FortiAuthenticator Captive Portal settings.

This section involves:

Creating the RADIUS client

1. Go to *Authentication* > *RADIUS Service* > *Clients*, and create a new RADIUS client.
2. Configure the following settings:

Name	Enter a name for the RADIUS client
Client name/IP	Enter the IP of the RADIUS client FortiGate
Secret	Enter the pre-shared secret. The same secret will be used to authenticate the FortiGate to the FortiAuthenticator.
Authentication method	Select *Password-only authentication (excludes users without a password)*
Realms	Select *local* | *Local users*
Enable captive portal	Select *Social portal (URL: /social_login/)*

Select *Save* and then *OK*.

Creating a user group

1. Go to *Authentication* > *User Management* > *User Groups*, and create a new user group.
2. Enter a name for the group. Users that log into any of the social portals will be placed in this group, once the group is added to the Captive Portal General Settings.

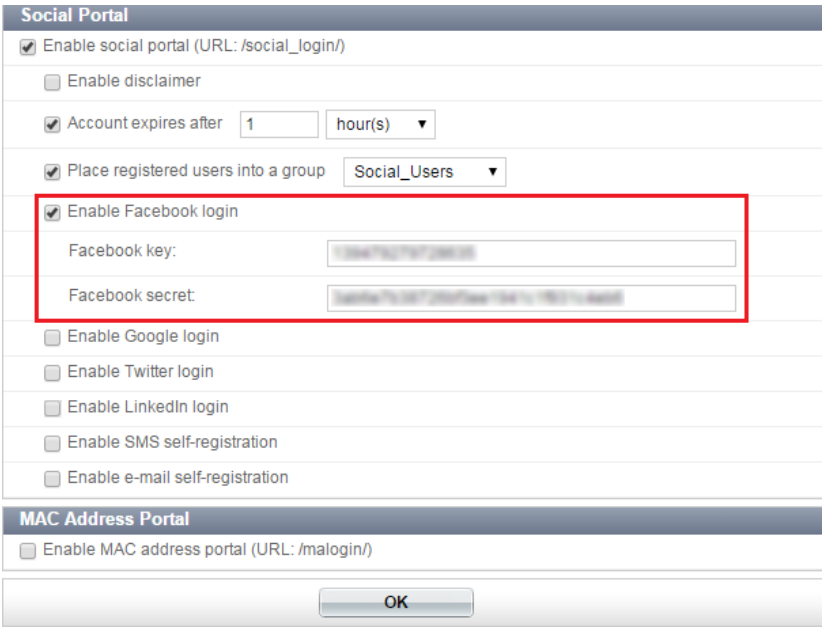
Configuring captive portal

1. Next go to *Authentication* > *Captive Portal* > *General*.
2. Enable *Social Portal*. The following options allow you to:
   - enable a disclaimer that users must accept before they are granted access to the Internet
   - set how long each social login will last for (hours, days, months, or years)

- set *Place registered users into a group* to the user group created previously
- to enable different social portal logins:
  - Facebook
  - Google
  - Twitter
  - LinkedIn
  - SMS self-registration
  - e-mail self-registration

Once enabled, each social network login requires a unique key and secret which can be viewed at any time from each of the social developer accounts; they are called the Client ID and Client Secret.

The following image shows an example Facebook login configuration:



Now that the FortiAuthenticator configuration is complete, you can proceed to the FortiGate configuration on page 26.

FortiGate configuration

With the desired social networks configured, and with FortiAuthenticator properly set up, the next step is to configure your FortiGate.

This section involves:

Configuring the RADIUS server

1. On the FortiGate, go to *User & Device* > *Authentication* > *RADIUS Servers* and select *Create New*.
2. Enter the following:

Name	Enter a name for the RADIUS server.
Primary Server IP/Name	Enter the IP address of the FortiAuthenticator.
Primary Server Secret	Enter the private key for the RADIUS server.

3. Select *Test Connectivity*.
4. Enter a valid *User* and *Password* and select *Test*.
   If your test successfully connects, proceed to the next step, Creating the RADIUS user group. Otherwise, review your RADIUS server configuration and, if necessary, double-check your FortiAuthenticator for the correct FortiAuthenticator configuration.

CLI Syntax

```
config user radius
   edit "FortiAuthenticator"
      set server "x.x.x.x"
      set secret ENC <snip>
   next
end
```

Creating the RADIUS user group

1. Go to *User & Device* > *User* > *User Groups.*
2. Create a new user group for "Social_Users".

3. Set the *Type* to *Firewall*.
4. Under *Remote groups*, select *Create New*.
5. For *Remote Server*, select the FortiAuthenticator RADIUS server configured above (see Configuring the RADIUS server).
6. Select *OK*.

CLI Syntax

```
config user group
   edit "Social_Users"
      set member "FortiAuthenticator"
   next
end
```

Configuring the WiFi security mode

1. Go to *WiFi & Switch Controller* > *WiFi Network* > *SSID* and edit the WiFi interface (alternatively, you can go to *System* > *Network* > *Interfaces* to edit the WiFi interface).
2. Under *WiFi Settings*, set the *Security Mode* to *Captive Portal* and set the *Portal Type* to *Authentication*.
3. Next to *Authentication Portal*, select *External* and enter the address for the captive portal.
   For example, `https://<FAC_FQDN>/social_login`.

Note that the URL for the external captive portal must include `/social_login`.

4. Set *User Groups* to the "Social_Users" group and select *OK*.

CLI Syntax

```
config wireless-controller vap
   edit "wifi"
      set vdom "root"
      set ssid "wifi"
      set security captive-portal
      set selected-usergroups "Social_Users"
      set local-switching disable
      set external-web "https://<FAC_FQDN>/social_login"
   next
end
```

Configuring exempt rules for social network authentication

To allow the user to authenticate to the social network sites before they are allowed to browse to the wider Internet, some exemptions are required. This can be a rather time-consuming process if you use only the GUI, so it might be best to enter the following exempt rules using the CLI.

Social network authentication exempt rules - CLI

Facebook

```
config firewall address
   edit "FB0"
      set subnet 5.178.32.0 255.255.240.0
   next
   edit "FB1"
      set subnet 195.27.154.0 255.255.255.0
   next
   edit "FB2"
      set subnet 80.150.154.0 255.255.255.0
   next
   edit "FB3"
      set subnet 77.67.96.0 255.255.252.0
   next
   edit "FB4"
      set subnet 212.119.27.0 255.255.255.128
   next
   edit "FB5"
      set subnet 2.16.0.0 255.248.0.0
   next
   edit "FB6"
      set subnet 66.171.231.0 255.255.255.0
   next
   edit "FB7"
      set subnet 31.13.24.0 255.255.248.0
   next
   edit "FB8"
      set subnet 31.13.64.0 255.255.192.0
   next
   edit "FB9"
      set subnet 23.67.246.0 255.255.255.0
   next
   edit "akamai-subnet-23.74.8"
      set subnet 23.74.8.0 255.255.255.0
   next
   edit "akamai-subnet-23.74.9"
      set subnet 23.74.9.0 255.255.255.0
   next
   edit
   "akamaihd.net"
      set type fqdn
      set fqdn "akamaihd.net"
   next
   edit "channel-proxy-06-frc1.facebook.com"
      set type fqdn
      set fqdn "channel-proxy-06-frc1.facebook.com"
   next
   edit "code.jquery.com"
      set type fqdn
      set fqdn "code.jquery.com"
   next
   edit "connect.facebook.com"
```

```
      set type fqdn
      set fqdn "connect.facebook.com"
    next
    edit "fbcdn-photos-c-a.akamaihd.net"
      set type fqdn
      set fqdn "fbcdn-photos-c-a.akamaihd.net"
    next
    edit "fbcdn-profile-a.akamaihd.net"
      set type fqdn
      set fqdn "fbcdn-profile-a.akamaihd.net"
    next
    edit "fbexternal-a.akamaihd.net"
      set type fqdn
      set fqdn "fbexternal-a.akamaihd.net"
    next
    edit "fbstatic-a.akamaihd.net"
      set type fqdn
      set fqdn "fbstatic-a.akamaihd.net"
    next
    edit "m.facebook.com"
      set type fqdn
      set fqdn "m.facebook.com"
    next
    edit "ogp.me"
      set type fqdn
      set fqdn "ogp.me"
    next
    edit "s-static.ak.facebook.com"
      set type fqdn
      set fqdn "s-static.ak.facebook.com"
    next
    edit "static.ak.facebook.com"
      set type fqdn
      set fqdn "static.ak.facebook.com"
    next
    edit "static.ak.fbcdn.com"
      set type fqdn
      set fqdn "static.ak.fbcdn.com"
    next
    edit "web_ext_addr_SocialWiFi"
      set type fqdn
      set fqdn "web_ext_addr_SocialWiFi"
    next
    edit "www.facebook.com"
      set type fqdn
      set fqdn "www.facebook.com"
    next
  end
```

Google+

```
  config firewall address
    edit "www.googleapis.com"
      set type fqdn
      set fqdn "www.googleapis.com"
    next
    edit "accounts.google.com"
```

```
      set type fqdn
      set fqdn "accounts.google.com"
   next
   edit "ssl.gstatic.com"
      set type fqdn
      set fqdn "ssl.gstatic.com"
   next
   edit "fonts.gstatic.com"
      set type fqdn
      set fqdn "fonts.gstatic.com"
   next
   edit "www.gstatic.com"
      set type fqdn
      set fqdn "www.gstatic.com"
   next
   edit "Google_13"
      set subnet 216.58.192.0 255.255.224.0
```

Accounts.google.com is too dynamic for an FQDN policy to function. This IP policy covers the whole range of possible subnets.

```
   next
end
```

Twitter

```
config firewall address
   edit "api.twitter.com"
      set type fqdn
      set fqdn "api.twitter.com"
   next
   edit "abs.twimg.com"
      set type fqdn
      set fqdn "abs.twimg.com"
   next
   edit "abs-0.twimg.com"
      set type fqdn
      set fqdn "abs-0.twimg.com"
   next
end
```

LinkedIn

```
config firewall address
   edit "www.linkedin.com"
      set type fqdn
      set fqdn "www.linkedin.com"
   next
   edit "api.linkedin.com"
      set type fqdn
      set fqdn "api.linkedin.com"
   next
   edit "static.licdn.com"
      set type fqdn
      set fqdn "static.licdn.com"
   next
   edit "help.linkedin.com"
      set type fqdn
      set fqdn "help.linkedin.com"
   next
```

```
    edit "www.fortinet.com"
       set type fqdn
       set fqdn "www.fortinet.com"
    next
end
```

Adding the social network authentication exempt rules to address groups

Add the exempt rules noted above to address groups for each social network. This can be done in the GUI under *Policy & Objects* > *Objects* > *Address* > *Create New Address Group*, or in the CLI as provided below.

Facebook

```
config firewall addrgrp
   edit "Facebook_Auth"
      set member "FB0" "FB1" "FB2" "FB3" "FB4" "FB5" "FB6" "FB7" "FB8" "FB9" "akamai-
          subnet-23.74.8" "akamai-subnet-23.74.9" "akamaihd.net" "channel-proxy-06-
          frc1.facebook.com" "code.jquery.com" "connect.facebook.com" "fbcdn-photos-c-
          a.akamaihd.net" "fbcdn-profile-a.akamaihd.net" "fbexternal-a.akamaihd.net"
          "fbstatic-a.akamaihd.net" "m.facebook.com" "ogp.me" "s-static.ak.facebook.com"
          "static.ak.facebook.com" "static.ak.fbcdn.com" "web_ext_addr_SocialWiFi"
          "www.facebook.com" "FortiAuthenticator"
   next
end
```

Google+

```
config firewall addrgrp
   edit "Google_Auth"
      set member "ssl.gstatic.com" "accounts.google.com" "www.googleapis.com"
          "fonts.gstatic.com" "www.gstatic.com" "Google_13"
   next
end
```

Twitter

```
config firewall addgrp
   edit "Twitter_Auth"
      set member "api.twitter.com" "abs.twimg.com" "abs-0.twimg.com"
   next
end
```

LinkedIn

```
config firewall addrgrp
   edit "LinkedIn_Auth"
      set member "api.linkedin.com" "www.linkedin.com" "help.linkedin.com"
          "www.fortinet.com" "static.licdn.com"
   next
end
```

Creating outbound social network authentication policies

The next step in the FortiGate configuration is to create outbound policies allowing access to each social network. In each case, it is imperative that you add the syntax: `set-captive-portal-exempt enable`. Note that this command is only available in the CLI, while everything else can be done in the GUI under *Policy & Objects > Policy > IPv4*.

1. On the FortiGate, go to *Policy & Objects > Policy > IPv4* and select *Create New*.

You may wish to create a separate outbound policy for each social network portal.

2. Set the following:

Incoming Interface	Set to the WiFi SSID.
Source Address	all
Outgoing Interface	Set to the Internet-facing interface.
Destination Address	• Facebook_Auth • Google_Auth • Twitter_Auth • LinkedIn_Auth
Schedule	always
Service	ALL
Action	ACCEPT

3. *Enable NAT* and select *OK*.
4. Click and drag this policy directly below the policy created for FortiAuthenticator access.
5. Go to *System > Dashboard > Status* and enter the following into the *CLI Console*:

```
config firewall policy
   edit 4 <-- assuming this is the correct policy ID
      set captive-portal-exempt enable
   next
end
```

CLI Syntax

```
config firewall policy
   edit 4 <-- assuming this is the correct policy ID
      set srcintf "wifi"
      set dstintf "wan1"
      set srcaddr "all"
      set dstaddr "Facebook_Auth" "Google_Auth" "Twitter_Auth" "LinkedIn_Auth"
      set action accept
```

```
        set schedule "always"
        set service "ALL"
        set captive-portal-exempt enable
        set nat enable
    next
end
```

Creating the FortiAuthenticator firewall object and access policy

The final steps in the Fortigate configuration is to create a policy that allows access to the FortiAuthenticator. You will have to add the FortiAuthenticator IP range as a firewall object before creating the policy.

Creating the FortiAuthenticator firewall object

1. Go to *Policy & Objects* > *Objects* > *Addresses* and select *Create New*.
2. Set the *Name* to "FortiAuthenticator".
3. Enter the FortiAuthenticator *Subnet / IP Range*.
4. Select *OK*.

CLI Syntax

```
config firewall address
   edit "FortiAuthenticator"
      set subnet x.x.x.x 255.255.255.255
   next
end
```

Creating the FortiAuthenticator access policy

Yet again, it is imperative that you add the following syntax to the policy: `set-captive-portal-exempt enable`. This command is only available in the CLI, while everything else can be done in the GUI under *Policy & Objects* > *Policy* > *IPv4*.

This policy rule is dependent on where the FortiAuthenticator is in relation to the client (not the FortiGate). In other words, the rule needs to originate from the client, with the FortiAuthenticator as its destination.

If the FortiAuthenticator is located on the internal interface, then the policy route is from WiFi to internal. If the FortiAuthenticator is located on the WAN, then the policy route is from WiFi to external.

1. On the FortiGate, go to *Policy & Objects* > *Policy* > *IPv4* and select *Create New*.
2. Set the following:

Incoming Interface	Set to the WiFi SSID.
Source Address	all

Outgoing Interface	Set to the Internet-facing interface.
Destination Address	FortiAuthenticator
Schedule	always
Service	ALL
Action	ACCEPT

3. *Enable NAT* and select *OK*.
4. Click and drag this policy to the top of the WiFi-to-External policy section.
5. Go to *System* > *Dashboard* > *Status* and enter the following into the *CLI Console*:

```
config firewall policy
   edit 3 <-- assuming this is the correct policy ID
      set captive-portal-exempt enable
   next
end
```

CLI Syntax

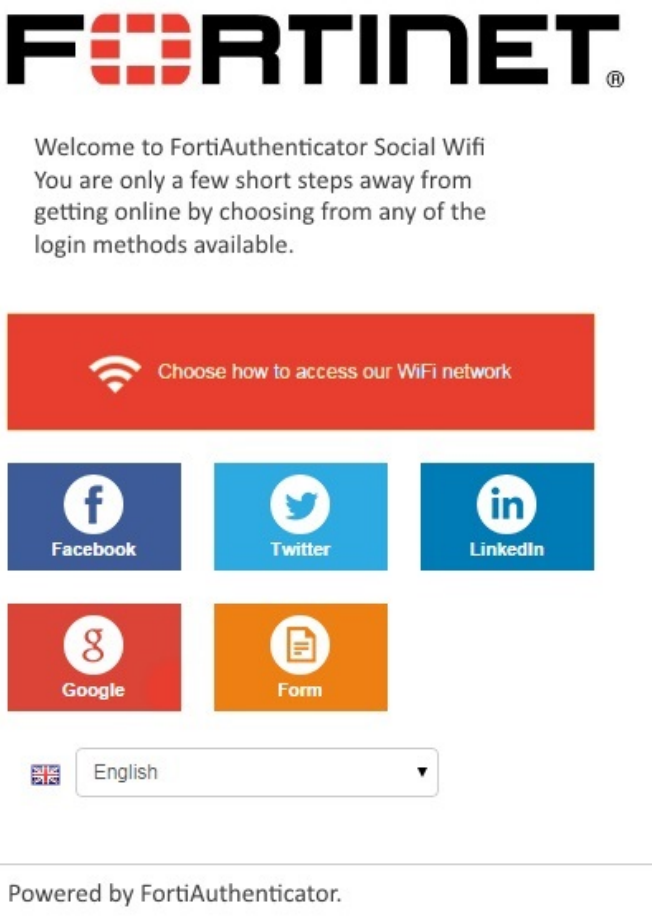
```
config firewall policy
   edit 3 <-- assuming this is the correct policy ID
      set srcintf "wifi"
      set dstintf "wan1"
      set srcaddr "all"
      set dstaddr "FortiAuthenticator"
      set action accept
      set schedule "always"
      set service "ALL"
      set captive-portal-exempt enable
      set nat enable
   next
end
```

Replacement Message configuration

FortiAuthenticator's default captive portal splash page includes links for each type of social network and authentication method. Using the following instructions you can customize a replacement message for the splash page to suit your configuration.

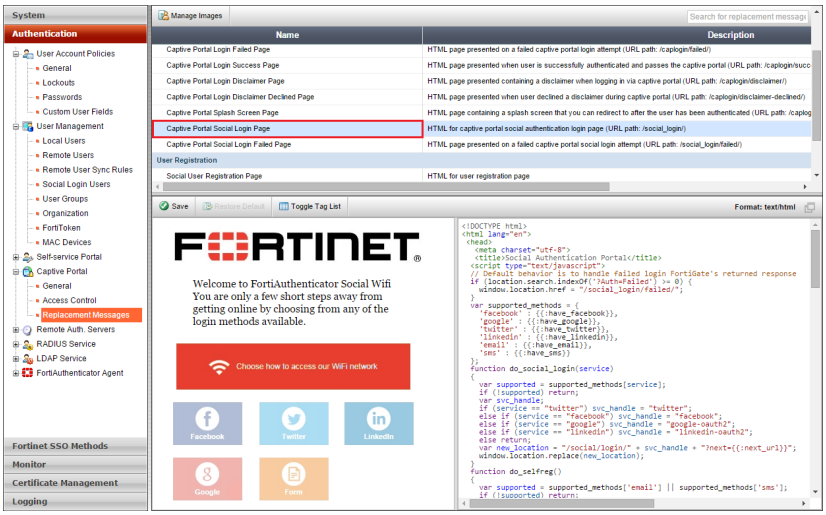
Editing the replacement message

Initially, the splash page appears as follows:



Depending on your FortiAuthenticator captive portal configuration, you might not have Twitter or Form-based authentication activated, and so you would want to remove these options from the splash page. To do so, follow the instructions below.

1. On the FortiAuthenticator, navigate to *Authentication* > *Captive Portal* > *Replacement Messages* and find the *Captive Portal Social Login Page* entry in the list.



2. In the *Format: text/html* panel, find the html code for the table that contains the social network links and delete the <tr> entries you do not wish to include.

   In the example below, we removed Twitter and Form-based authentication:

```
<table id="main_table">
<tr>
   <td colspan=3><div id="fortinet_logo">  </div></td>
</tr>
<tr>
   <td colspan=3>
   <div id="welcome_text" style="text-align:left">
      Welcome to FortiAuthenticator Social Wifi <br>
      You are only a few short steps away from <br>
      getting online by choosing from any of the <br>
      login methods available. <br>
   </span>
   </td>
</tr>
<tr>
   <td colspan=3><div id="login_banner" title="Choose how to access our wifi network">
        </div></td>
</tr>
<tr>
   <td><div id="facebook_btn" class="login_btn" onclick='return do_social_login
       ("facebook");' title="Login with Facebook">  </div></td>
   <td><div id="linkedin_btn" class="login_btn" onclick='return do_social_login
       ("linkedin");' title="Login with LinkedIn">  </div></td>
   <td><div id="google_btn" class="login_btn" onclick='return do_social_login
       ("google");' title="Login with Google">  </div></td>
</tr>
</table>
```

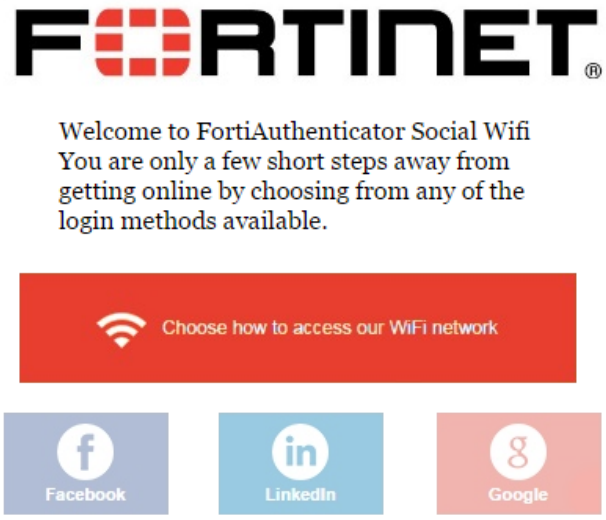
If you make a mistake, you can use *Ctrl+Z* to undo your mistake. Also, you can select *Restore Default* to return to the default splash page configuration.

3. To improve the style of the splash page (changing the spacing and such), you will need to modify the stylesheet. In the following example, the `margin-left` syntax was modified to reposition the images:

```
#facebook_btn {
   background: url("{{:image/social_facebook}}") no-repeat scroll center center #FFFFFF;
   margin-left: 15px;
   padding-bottom: 30px;
   opacity: 0.4;
}
#linkedin_btn {
   background: url("{{:image/social_linkedin}}") no-repeat scroll center center #FFFFFF;
   margin-left: 20px;
   opacity: 0.4;
}
```

4. Select *Save*.

Following the instructions above, your splash screen should appear as below:



De-Authenticating users

Although you configure account expiry in the FortiAuthenticator social portal settings, for various reasons you may wish to forcefully de-authenticate users prior to the expiry time. The following steps,involving both the FortiGate and the FortiAuthenticator, demonstrates how to forcefully de-authenticate users.

1. On the FortiGate, go to *User & Device* > *Monitor* > *Firewall* and select the user(s) you wish to remove.
2. Select *De-authenticate*. When prompted, select *OK*.
3. On the FortiAuthenticator, go to *Authentication* > *User Management* > *Social Login Users* and select the same user(s) as in step 1.
4. Select *Delete*. When prompted, select *Yes, I'm sure*.

   The user(s) can no longer browse the Internet through the FortiGate.

Note, however, that session timeouts may still apply.